



## 经验谈

# 基于MSP430单片机的虚拟I<sup>2</sup>C总线软件包

阅读本刊所载的何立民教授《按平台模式设计的虚拟I<sup>2</sup>C总线软件包VIIC》(简称何文)一文后,深感可以将这一方法推广到别的单片机上,这样使用户可利用通用I/O口的模拟,随意规定虚拟I<sup>2</sup>C总线端口,扩大了虚拟I<sup>2</sup>C总线应用的灵活性。根据文中所提出的前归一化和后归一化等设计原则,本人设计了基于MSP430单片机的虚拟I<sup>2</sup>C总线软件包VIIC\_M1.0。

## 一、VIIC\_M1.0软件包的组成

根据归一化设计的要求,主方式下虚拟I<sup>2</sup>C总线由下列10个子程序组成:

### ① 时序模拟子程序4个:

I2C\_Sta, I2C\_Stop, I2C\_Mack, I2C\_Mnack

### ② 操作模拟子程序3个:

I2C\_Ackn, I2C\_WR\_Byte, I2C\_RD\_Byte

### ③ 数据读写子程序3个:

I2C\_WR\_NByte, I2C\_RD\_NByte, I2C\_WR\_Addr

限于篇幅,软件包的程序清单(包括正文)已发至本刊网站,可下载。

## 二、应用对象

### 1. MSP430单片机

MSP430单片机为低功耗的16位单片机,有MSP430×11x、MSP430×112、MSP430×1101、MSP430×13x、MSP430×14x、MSP430×31x、MSP430×32x、MSP430×33x等型号,每种芯片都有丰富的I/O端口。

本设计所使用的芯片为MSP430E325。

#### (1) 虚拟I<sup>2</sup>C总线所使用的I/O端口

① 数据线(SDA)使用的是通用端口P0中的P0.7。该端口为输入/输出双向口,有输入寄存器(P0IN)、输出寄存器(P0OUT)及方向寄存器

(P0DIR)等寄存器,通过字节指令访问。

② 时钟线(SCL)使用的是通用定时器/端口TP中的TP.1。该端口为输出口,有定时器/端口控制寄存器(TPCTL)、定时器/端口数据寄存器(TPD)及定时器/端口允许寄存器(TPE)等,通过字节指令访问。

#### (2) 运行时所使用的时钟频率

MSP430E325运行时用两个时钟:辅助时钟(ACLK)和主时钟(MCLK)。当采用32768kHz的晶体振荡器,并且系统时钟控制寄存器采用缺省值时,主时钟的频率为1.049MHz。

若主时钟的频率不是1.049MHz,请适当调整程序中nop的个数。

## 2. EEPROM器件

### (1) EEPROM器件

本例使用的EEPROM器件为24LC65(MICROCHIP)。24LC65(MICROCHIP)容量为8KB,其中的字节地址为13位,分两个字节:

Subaddr\_H为高5位地址;

Subaddr\_L为低8位地址。

### 24LC65(MICROCHIP)

的封装引脚如图1所示。

### (2) 24LC65的数据格式

式

#### ① 当前地址写:

S,SLA+W,A,DATA1,

A,DATA2,A,...,DATAn,A,P

#### ② 当前地址读:

S,SLA+R,A,DATA1,A,DATA2,A,...,DATAn/A,P

#### ③ 指定地址写:

S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,A,DATA1,

A,DATA2,A,...,DATAn,A,P

#### ④ 指定地址读:

S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,A,

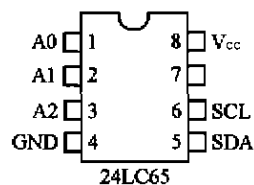


图1 24LC65的封装引脚

S,SLA+R,A,DATA1,A,DATA2,A,...,DATAn,/A,P

其中:

SLA+W,SLA+R,Subaddr\_H,Subaddr\_L 为主控器件发出的数据

S,A,/A,P 为主控器件发出的信息

DATA1,DATA2,...,DATAn 为被控器件发给主控器件的数据

A 为被控器件发给主控器件的信息

(3) 24LC65 与 MSP430-325 的连接

24LC65 与 MSP430-325 的连接如图 2 所示。

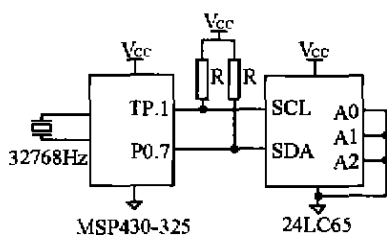


图 2 MSP430 与 EEPROM 24LC65 连接示意图

### 三、应用界面

#### 1. 发送 N 字节数据——从当前地址开始

;发送的数据在 MTD 中

;

;数列格式: S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,  
A,DATA1,A,DATA2,A,...,DATAn,A,P

;

MOV.b #CODE,SLA ;I2C\_R\_R/W=0,  
;A0=A1=A2=0

MOV.b #N,Num\_byt ;发送字节数

CALL I2C\_WR\_Nbyte ;从当前地址开始写

#### 2. 接收 N 字节数据——从指定地址开始读

;接收的数据在 MRD 中

;

;数列格式:S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,  
A,

S,SLA+R,A,DATA1,A,DATA2,A,...,DATAn,/A,P

MOV.b #CODE+SLAR/W,SLA  
;SLAR/W=1,A0=A1=A2=0

MOV.b #N,Num\_byt ;接收字节数

CALL #I2C\_RD\_Nbyte ;从当前地址开始读

#### 3. 接收 N 字节——从指定地址开始读

;接收的数据存放在 MRD 中

;

;数列格式:S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,

A,

; S,SLA+R,A,DATA1,A,DATA2,A,...,DATAn,/A,P

;

MOV.b #CODE,SLA

;SLAR/W=0;写,A0=A1=A2=0

MOV.b #N,Num\_byt ;接收字节数

MOV #Subaddr,I2C\_R\_Addr

;字节地址送入暂存单元

CALL #I2C\_WR\_Addr ;发送字节地址

MOV.b #CODE+SLAR/W,SLA

;SLAR/W=1;读,A0=A1=A2=0

CALL #I2C\_RD\_Nbyte

#### 4. 发送 N 字节——从指定地址开始写

;数列格式: S,SLA+W,A,Subaddr\_H,A,Subaddr\_L,

A,DATA1,A,DATA2,A,...,DATAn,A,P

;

MOV.b #CODE,SLA

;SLAR/W=0;写,A0=A1=A2=0

MOV.b #N,Num\_byt ;发送字节数

MOV #Subaddr,I2C\_R\_Addr

;字节地址送入暂存单元

CALL #I2C\_WR\_Addr ;发送字节地址

CALL #WR\_Nbyte

后记:

① 如果使用 MSP430 中别的 I/O 端口,只需要在模拟时序子程序中改动相应的寄存器即可。本例中时钟线使用 TP 口,是因为它为输出口,在程序中可以节省指令。

对于 24C 系列的其它芯片,由于容量不同,请特别注意其字节地址是 1 个字节还是 2 个字节。本例的 24LC65 字节地址就是 2 个字节的。

② MSP430 系列的单片机本人是初学使用,程序虽经初步调试通过,轻易抛出,主意为引玉之砖,作为学习何文的一点体会。缺点和错误在所难免,望各位不吝指教。

(北京工业大学 张俊谟)